Fig.1
(PRIOR ART)

TRANSLATION
TABLE MEMORY,
207

IPA	ENA

TTE
211(i)

209(u)

TT,209(a)

QUEUE
MEMORY,
213

DESC.
217(I)

QT,
221

217(o) | B T

QH,
219

QUEUE,215(A)

208

SERIAL
PAIR,
205(0)

204

206

DCPIC,
203

TO:FROM
DIGITAL
SWITCH,221

TO:FROM
GPIO,223

HOST,
227

225

205(15)

228

BT,
233

BUFFER,
231(n)

BUFFER
MEMORY,
229

BUFFER,
231(a)

201

Fig.2

Fig.3

Fig.4

```
┌─────────────────────────────┐  ┐
│  SYSTEM INTERFACE CONFIG.    │  │  XP
│  INFO, 521                   │  │  LOCAL MEMORY,
├─────────────────────────────┤  │  517
│                             │  │
│  GLOBAL CONFIG. REGS., 519   │  ┘
├─────────────────────────────┤  ┐  BME
│                             │  │  LOCAL
│  BUFFER STATUS, 515          │  │  MEMORY,513
├─────────────────────────────┤  ┘
│  MAILBOXES, 511              │  ┐  QME
├─────────────────────────────┤  │  LOCAL
│                             │  │  MEMORY,507
│  QUEUE STATUS, 509           │  ┘
├─────────────────────────────┤
│  FP LOCAL MEMORY, 505        │
├─────────────────────────────┤
│  501(15)                     │
├─────────────────────────────┤
```

MEMORY,
504
FOR
CHANNEL
PROCESSORS,
307(0..15)

```
┌─────────────────────────────┐
│  501(3)                      │  ┐
├─────────────────────────────┤  │
│  501(2)                      │  │  CLUSTER
├─────────────────────────────┤  │  MEMORY,
│  501(1)                      │  │  503(0)
├─────────────────────────────┤  │
│  CHANNEL                     │  │
│  PROCESSOR LOCAL MEMORY      │  │
│  501(0)                      │  ┘
└─────────────────────────────┘
```

321

Fig.5

Replacement Sheet

DMEM, 405

EVENT REGS {
DATASCOPE EVENT REGS, 632 → 632(0)

GEN'L EVENT REGS, 630
}

CYCLE COUNTER, 619

EVENT TIMER, 629

QUEUE STATUS, 621

SDP MODE, 627

CP MODE, 625

TxSDP CTL, 615

RxSDP CTL, 613

SONET OVERHEAD BITS, 612

RINGBUS CONTROL REGS, 617

611 {
612(1)

RdCB, 612(0)

610(1)

WrCB, 610(0)
}

624(1)

DATA SCOPE REGS 624(0)

DATA SCOPE, 625(0)

Tx STATUS, 639   } Tx DATA SCOPE, 641

Tx CB, 637

MERGE REGS, 603

Rx STATUS, 635   } Rx DATA SCOPE, 643

Rx CB, 633

EXTRACT REGS, 601

601(i)

Fig.6

START — 703

RxSDP, 421

Rx DATA
(SDP Rx) — 705

A
707 — TRANSFER
PAYLOAD
TO SDRAM

B
LAUNCH TLE
LOOKUPS
ON RING BUS

709

C
EXTRACT AND
WRITE FIELDS TO
EXTRACTION
REGISTER — 711

STORED IN MEMORY
INTERFACE BETWEEN
RISC CORE AND SDP

CPRC, 401

713 — PROCESS, VALIDATE
AND CLASSIFY
CELL/FRAME FOR
FORWARDING

715 — ENQUEUE
DESCRIPTOR TO
QME

QME

305

XP QUEUE
(HOST)

CP QUEUE
(LOCAL FWD)

FP QUEUE
(REMOTE FWD) — 721

717

719

701

Fig.7

START

805 — RISC CORE DETECTING DATA TO TRANSMIT FROM QUEUE MGMT ENGINE VIA LOCAL STATUS

803

807 — DATA READY FOR TX IN QME ?   NO

YES

809 — REQUEST DESCRIPTOR FROM QME

811 — PASS ARGUMENTS TO SDP VIA MERGE REGISTER   RISC CORE WRITES INFORMATION TO THE CP MERGE REGISTER

CPRC, 401

813 — INITIATE PAYLOAD TRANSFER FROM SDRAM CTL TO SDP   RISC CORE WRITES INFORMATION TO THE CP CONTROL REGISTER

815 — INITIAL SDP TRANSMIT   RISC CORE DETECTS THE END OF FRAME

817 — FREE RESOURCES

801

Fig.8

OWN, 935	L5:L0 937	BUSY, 941

<u>635</u>

BTAG, 933			
OFFSET, 931			
Av, 929	NR, 927	Err, 925	OWN, 921
SDPST, 915	EOP, 927	BCTL, ST 919	
LENGTH, 911			
BUFFER POOL NO, 909			
DMEM DMAADDR, 907			
TxRcy ADDR., 905			
RxRcy ADDR., 903			
DMEM BYTE ADDR., 901			

RxCBCTL, 913

<u>633</u> Fig.9

Replacement Sheet

10/47

Fig.10

TO 417, 1028, 1025

1123

CRCS, 1111

COUNTER AND GENERAL REGS., 1115

ALU, 1119

CAM, 1117

NEXT ADDRESS LOGIC, 1105

ADDR.

CONTROL STORE, 1101

1121

EXT TEST

FROM ALU

FROM COUNTERS

1013

Fig. 11

SCI0981TS

12/47

Fig. 12

Fig. 13

MEM, 503(i,3)   503(i,2)

SDP, 1403	DMEM, 405	BUS CONTROL, 1405
RISC CORE, 1407	IMEM, 403	

BUS CONTROL	DMEM	SDP
	IMEM	RISC CORE

SIMEM, 1409

RISC CORE	IMEM	
SDP	DMEM	BUS CONTROL

	IMEM	RISC CORE
BUS CONTROL	DMEM	SDP

503(j,0)   503(j,1)

509

Fig. 14

Fig. 15

Fig. 16

CORE | FP RCLK
CLK |

FIFO, 1708 → 1619

1614 ← FABRIC HDR. INTERP., 1707

HEADER-PAYLOAD SEP., 1703

FIFO, 1705 → 1616

HDR. EXTR + INTERP., 1701

1614   1616

1617

CORE | FP TCLK
CLK |

FIFO, 1717 → 1623

FABRIC HDR. GEN., 1715

HDR. + PAYLOAD MERGE, 1711

FIFO, 1713 ← 1620

HDR. GEN., 1709

1618

1621

Fig. 17

FABRIC HEADER, 1803

FRAME HEADER, 1805

PAYLOAD, 1807

$32 \leq FL,\ 1809 \leq 128$

FABRIC FRAME, 1801

Fig. 18

PORT 1
PORT 2

PORT 16

DCP,
203(1)

SDRAM — 229(1)

SRAM — 207(1)

204(1,16),
206(1,16)

FP — 1903

PORT 17
PORT 18

PORT 32

DCP,
203(2)

SDRAM — 229(2)

SRAM — 207(2)

1905

204(2,16),
206(2,16)

MEMORY — 229(1)

204(1,1),
206(1,1)

PORT 1
PORT 2

PORT 16

DCP,
203(1)

303(1)

FP

SWITCHING
FABRIC

X

222

FP

DCP,
203(X)

FP

PORT 16 x n

303(X)

1901

MEMORY — 229(X)

Fig. 19

204(1,1),
206(1,1)

MEMORY — 229(1)

PORT 1
PORT 2

DCP,
203(1)

303(1)

FP

SWITCHING
FABRIC

X

222

CUSTOMER
SPECIFIC LINE
INTERFACES,
2003

APPLICATION
SPECIFIC LOGIC,
2002

FP

FP

2001

Fig. 20

ALG#2125

2124

ALG. DESC., 2123

VTNS, 2127 | ALG. SPEC, 2129

209

TP, 2101(0)

TP, 2101(1)

2105

LINK TABLE, 2107 (2106)

LIX, 2109 → LTE, 2111

TP, 2101(7)

DATA TABLE, 2117

DIX, 2114 → DTE, 2119

(2106)

CTL INFO, 2113 | LINK INFO, 2115

2111

207

KEY, 2120 | DATA, 2121

2119

Fig. 21

Fig. 22

301

| LVT1, 2325 | LVT2, 2327 | LVT3, 2329 | ✕ | HASHF#, 2331 | ✕ | DVT, 2333 |

ALGO#, 2125 →   ACR, 2301(i)   ACRS, 2301

| TYPE, 2335 | ✕ | SIZE, 2337 | BADDR, 2339 |

PT#, 2343 →   CTCR, 2311(i)   CTCRS, 2311

VT#, 2127 →

| | | | PT#, 2343 |

VTCR, 2341(i)   VTCRS, 2341

MESSAGE INFO, 2321

PROCESSING INFO, 2327

RESULT, 2329

MCR, 2319(K)   MESSAGE CONTEXT REGISTERS, 2319

2205

HASHF#, 2331 →

HASH CODE, 2323(i)

2215

Fig. 23

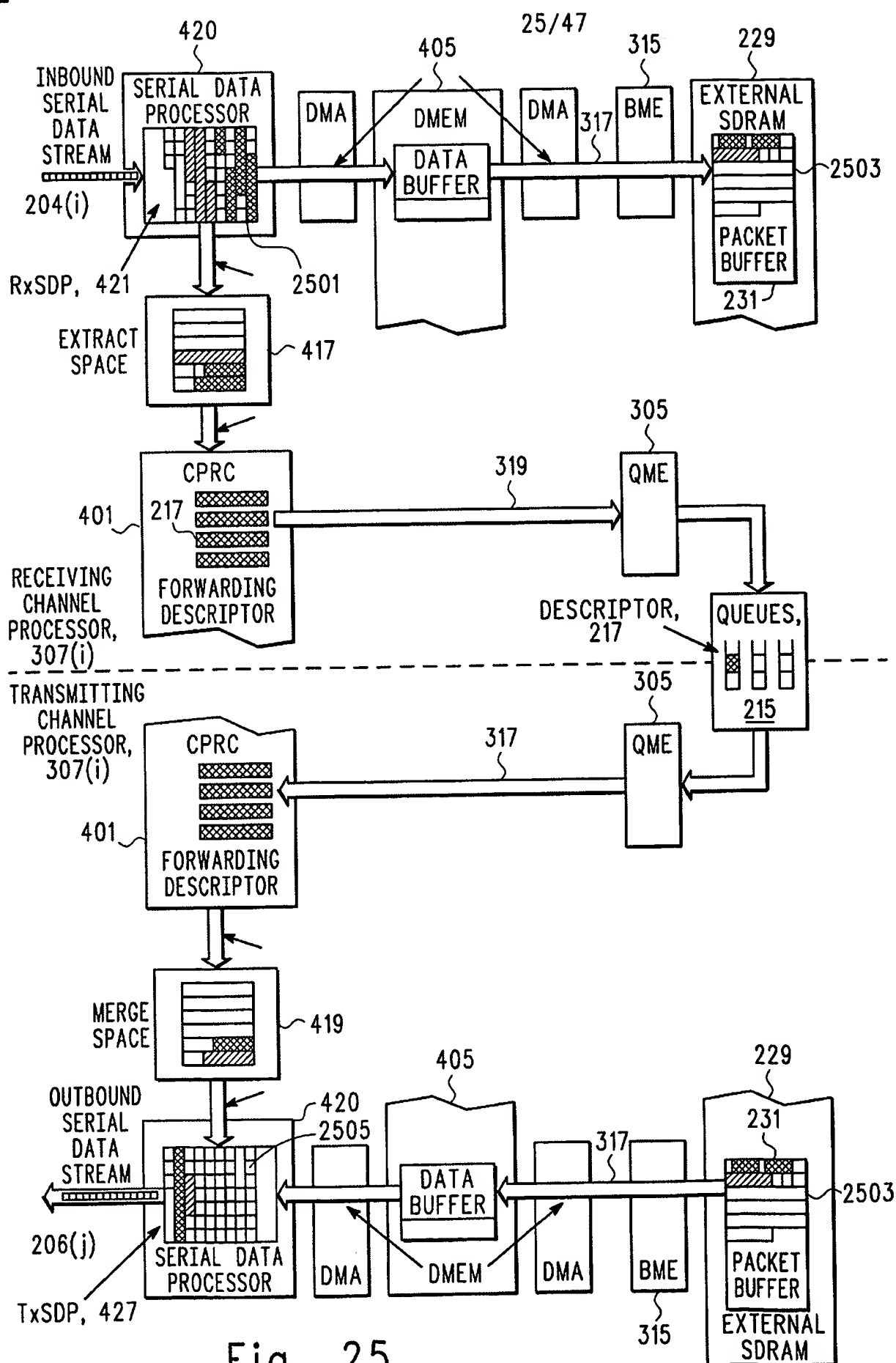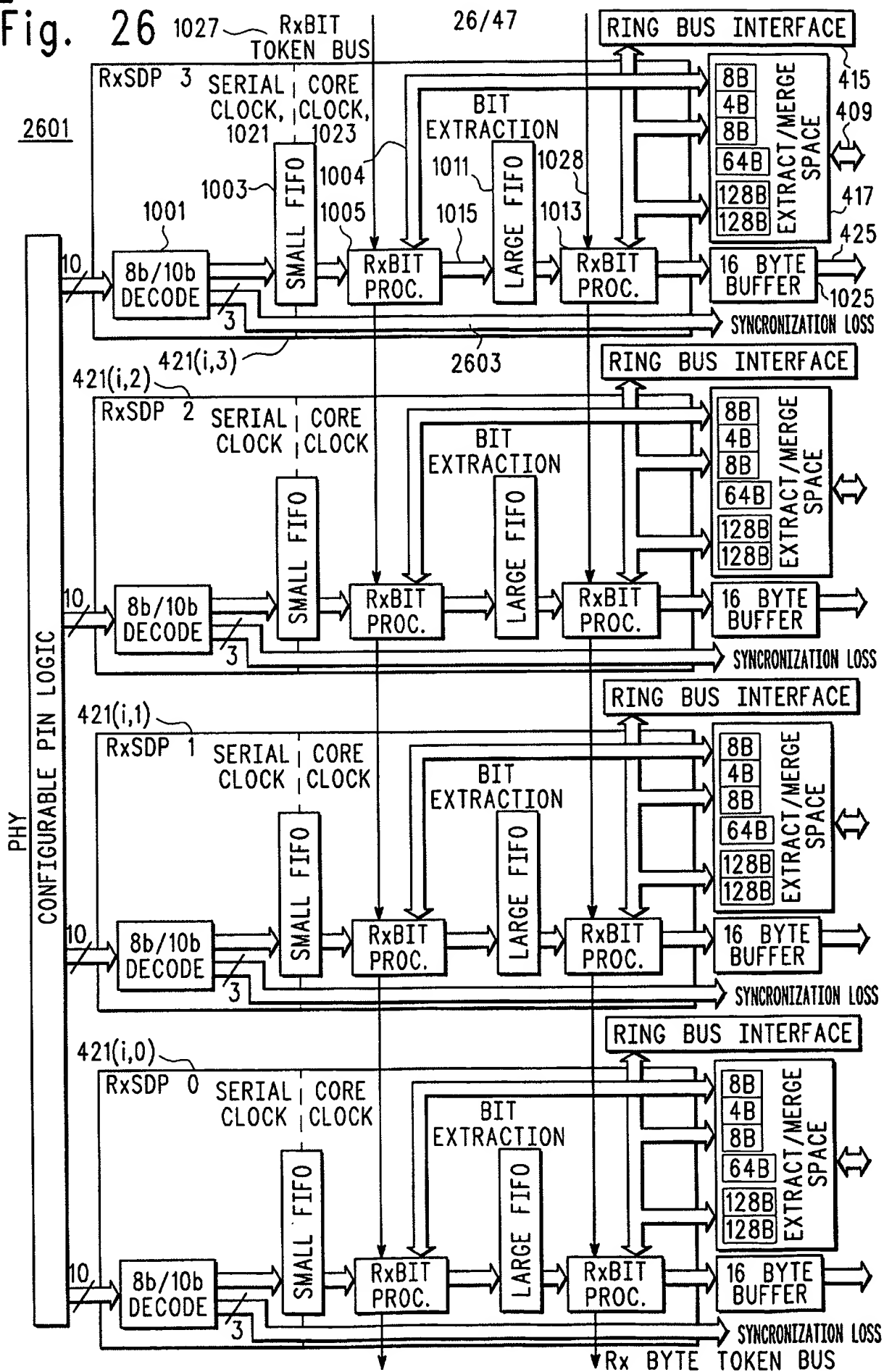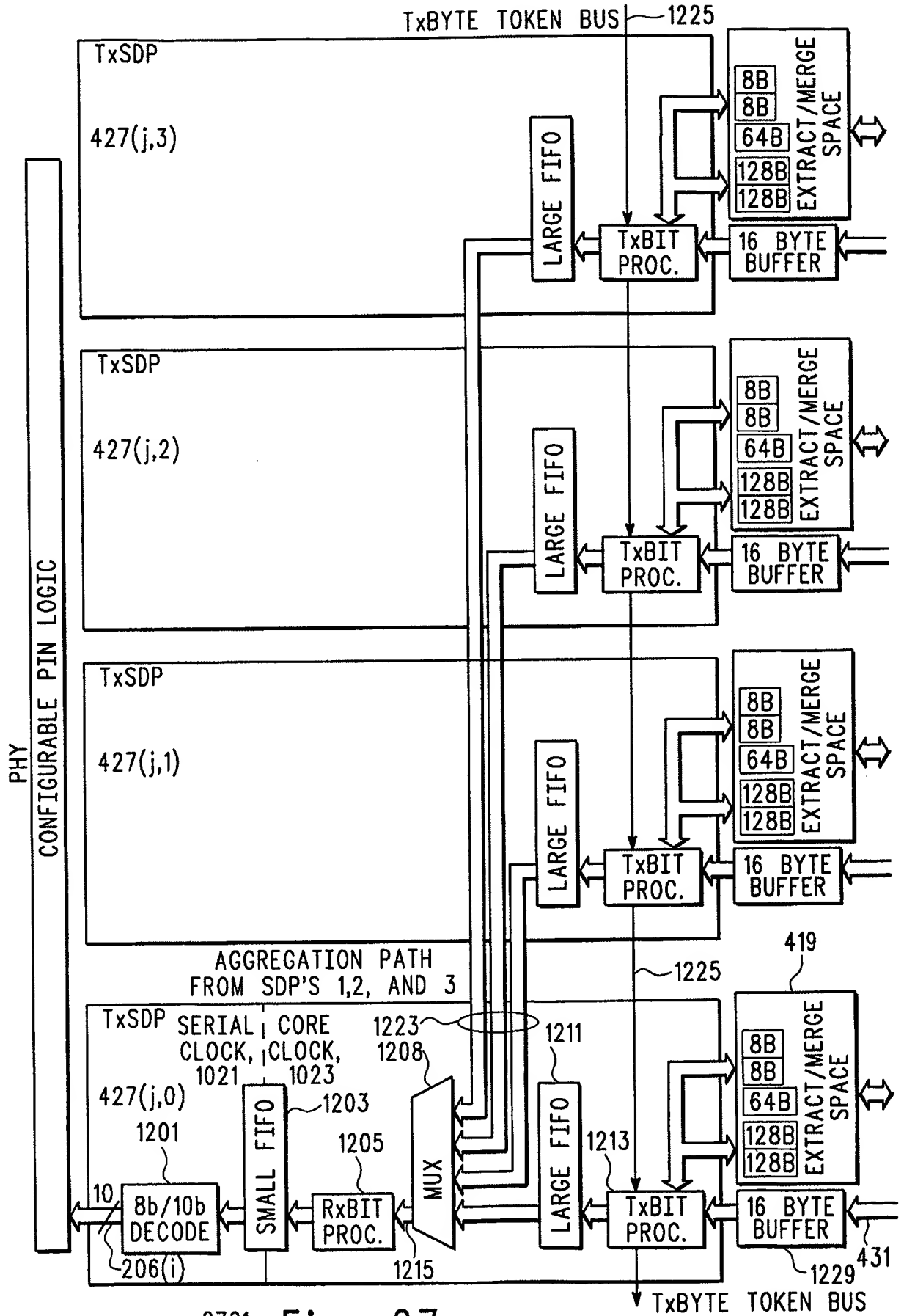| COMMAND	COMMAND ID	RETURN DATA	DESCRIPTION	
WRITE(VTABLE#,INDEX, MASK,DATA,OFFSET, LENGTH)	0X2	NONE	WRITE DATA INTO A VIRTUAL TABLE AT INDEX.	2401
READ(VTABLE#,INDEX, OFFSET,LENGTH)	0X3	DATA	READS DATA FROM A VIRTUAL TABLE.	2403
FINDW(ALG#,KEY)	0X6	PHYSICAL TABLE, INDEX, ERROR	FINDS A KEY USING ALG#. SETS RING BUS ERROR FLAG IF KEY IS NOT FOUND.	2405
FINDW(ALG#,KEY,DATA, OFFSET,LENGTH)	0X4	PASS/FAIL, INDEX, ERROR	WRITES DATA INTO A TABLE USING A KEY. SETS RING BUS ERROR FLAG IF THE KEY IS NOT FOUND.	2407
FINDR(ALG#,KEY,DATA, OFFSET,LENGTH)	0X5	PASS/FAIL, INDEX, DATA	READS LENGTH DWORDS OF DATA FROM A VTABLE# USING A KEY AT OFFSET DWORDS. SETS RING BUS ERROR FLAG IF THE KEY IS NOT FOUND	2409
XOR(VTABLE#,INDEX, DATA/PCRC,OFFSET, MASK,CRC,LAST)	0X1	NONE OR CRC IN CRC MODE	XOR'S UP TO A 32 BIT VALUE TO OFFSET. ONLY MASKS OF UP TO FOUR CONSECUTIVE BYTES ARE VALID. A SPECIAL MODE EXISTS FOR CRC CALCULATIONS.	2411
ADD(VTABLE#,INDEX, DATA,OFFSET,MASK)	0X7	NONE	ADDS UP TO A 32-BIT VALUE TO OFFSET. ONLY MASKS OF UP TO FOUR CONSECUTIVE BYTES ARE VALID.	2413
WRITEREG(REG-ADDR, DATA)	0X0,0x10	NONE	WRITE DATA TO TLE REGISTER AT REG_ADDR.	2415
READREG(REG_ADDR, DATA)	0X0,0x11	DATA	READ DATA FROM TLE REGISTER AT REG_ADDR.	2417
ECHO(DATA)	0X0,0x04	DATA	RETURN DATA FROM TLE FOR TEST PURPOSES	2419
NOP()	0X0,0x05	NONE	INSERTS A NOP INTO THE TLE PIPE	2420

2421 (groups 2401, 2403)
2423 (groups 2405, 2407, 2409)
2425 (groups 2411, 2413)
2427 (groups 2415, 2417)

Fig. 24

INBOUND SERIAL DATA STREAM

420

SERIAL DATA PROCESSOR

204(i)

RxSDP, 421

2501

405

DMA   DMEM   DMA   317   BME

DATA BUFFER

315

229

EXTERNAL SDRAM

2503

PACKET BUFFER

231

EXTRACT SPACE

417

RECEIVING CHANNEL PROCESSOR, 307(i)

401

CPRC

217

FORWARDING DESCRIPTOR

319

305

QME

DESCRIPTOR, 217

QUEUES, 215

TRANSMITTING CHANNEL PROCESSOR, 307(i)

CPRC

401

FORWARDING DESCRIPTOR

317

305

QME

MERGE SPACE

419

OUTBOUND SERIAL DATA STREAM

206(j)

420

2505

SERIAL DATA PROCESSOR

TxSDP, 427

DMA

405

DATA BUFFER

DMEM   DMA   317   BME

315

229

231

PACKET BUFFER

2503

EXTERNAL SDRAM

Fig. 25

Replacement Sheet

Fig. 26

BRIGHTMAN ET AL.
SCI0981TS

TxBYTE TOKEN BUS — 1225

TxSDP

427(j,3)

LARGE FIFO

TxBIT PROC.

16 BYTE BUFFER

8B
8B
64B
128B
128B
EXTRACT/MERGE SPACE

TxSDP

427(j,2)

LARGE FIFO

TxBIT PROC.

16 BYTE BUFFER

8B
8B
64B
128B
128B
EXTRACT/MERGE SPACE

TxSDP

427(j,1)

LARGE FIFO

TxBIT PROC.

16 BYTE BUFFER

8B
8B
64B
128B
128B
EXTRACT/MERGE SPACE

AGGREGATION PATH
FROM SDP'S 1,2, AND 3

1225

419

PHY
CONFIGURABLE PIN LOGIC

TxSDP SERIAL CORE
CLOCK, CLOCK,
1021   1023

1223
1208

1211

427(j,0)
1201

1203

1205

10
8b/10b DECODE

SMALL FIFO

RxBIT PROC.

MUX

LARGE FIFO

1213

TxBIT PROC.

16 BYTE BUFFER

8B
8B
64B
128B
128B
EXTRACT/MERGE SPACE

206(i)

1215

1229 431

TxBYTE TOKEN BUS

2701 Fig. 27

CTL, 2803 {	2805	2807	2809	2811	2813	2815
	M	TY	LEN	SEQ	DEST	SRC
	DATA, 2817					

RING BUS MESSAGE, 2801

Fig. 28

QME, 305

QSI IN GMEM, 2902

QMB, 2903(0) · · · QMB, 2903(n)

TQSR, 2905

DEMQ, 2907

2909

QOS, 2911

QCMD, 2913

RQSR, 2915

601

DEQM, 2907

CP307(i)

319

317

GB

PB

2901

Fig. 29

Replacement Sheet

3003	3005	3007	3009		3011
CNT	T#	PI	OP	✕	QUEUE #

QUEUE INST. ADDR., 3001

		3015		3017	3019
✕		DESCR. LIMIT	✕	DESCR. POOL	DESCR. ALLOW

CONFIGURE QUEUE WRITE DATA., 3013

3023

	DW	

DESCRIPTOR, 217	

⎫ 3021

UNICAST ENQUEUE DATA

	DW, 3023		QW, 3027		QL, 3029

DESCRIPTOR, 217	

⎫ 3025

UNICAST DEQUEUE DATA

3003	3005	3007	3009		
CNT	T#	PI'	OP		QLEV, 3033
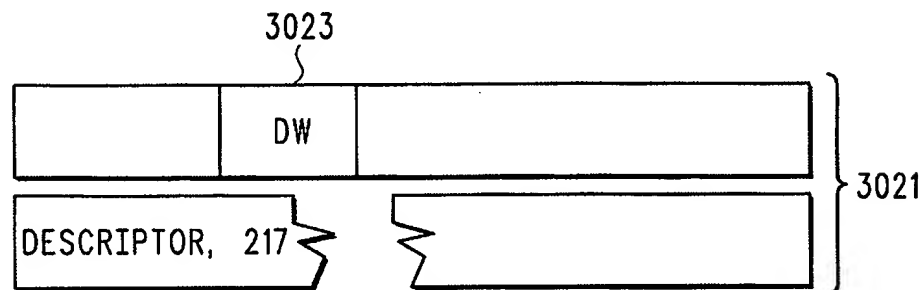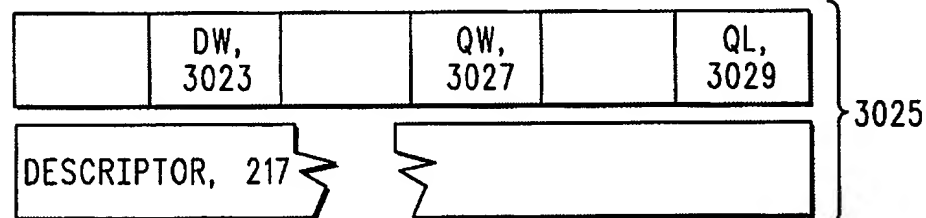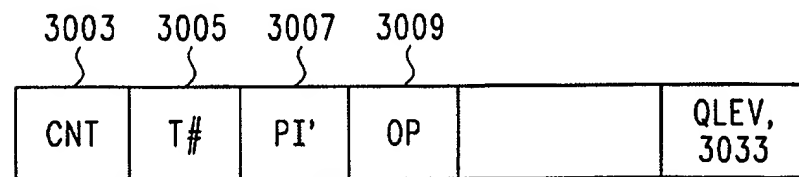
MULTICAST ENQUEUE INST. ADDR., 3031

	MCV, 3037			DW, 3023

DESCRIPTOR, 217	

⎫ 3035
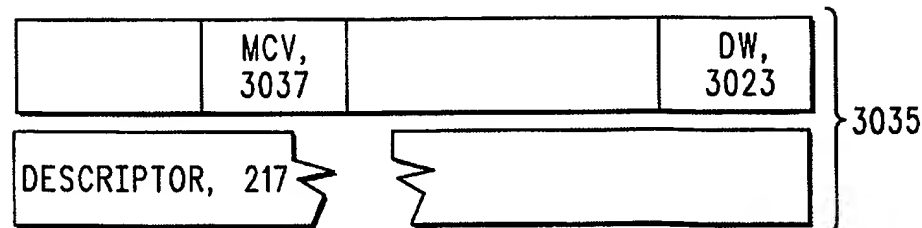
MULTICAST ENQUEUE DATA

2913

Fig. 30

Fig. 31

BRIGHTMAN ET AL.
SCI0981TS

UNICAST QUEUE, 215(i)

QR, 3103(i) ←····→ DR, 3111(a)   DR, 3111(b) ←····→ DR, 3111(c)

3115

3211

MLR, 3203 →HPTR, 3207→ DR, 3111 (g)   DR, 3111 (b)   DR, 3111 (i)

MCLR, 3123

3115

MCLR, 3123 (r)

3212

MULTICAST
LIST,
3201

3209

3213

MCLR, 3123 (s)

3115

3214

QR, 3103(j) →····→ DR, 3111 (k)   DR, 3111 (l) →····→ DR, 3111 (m)
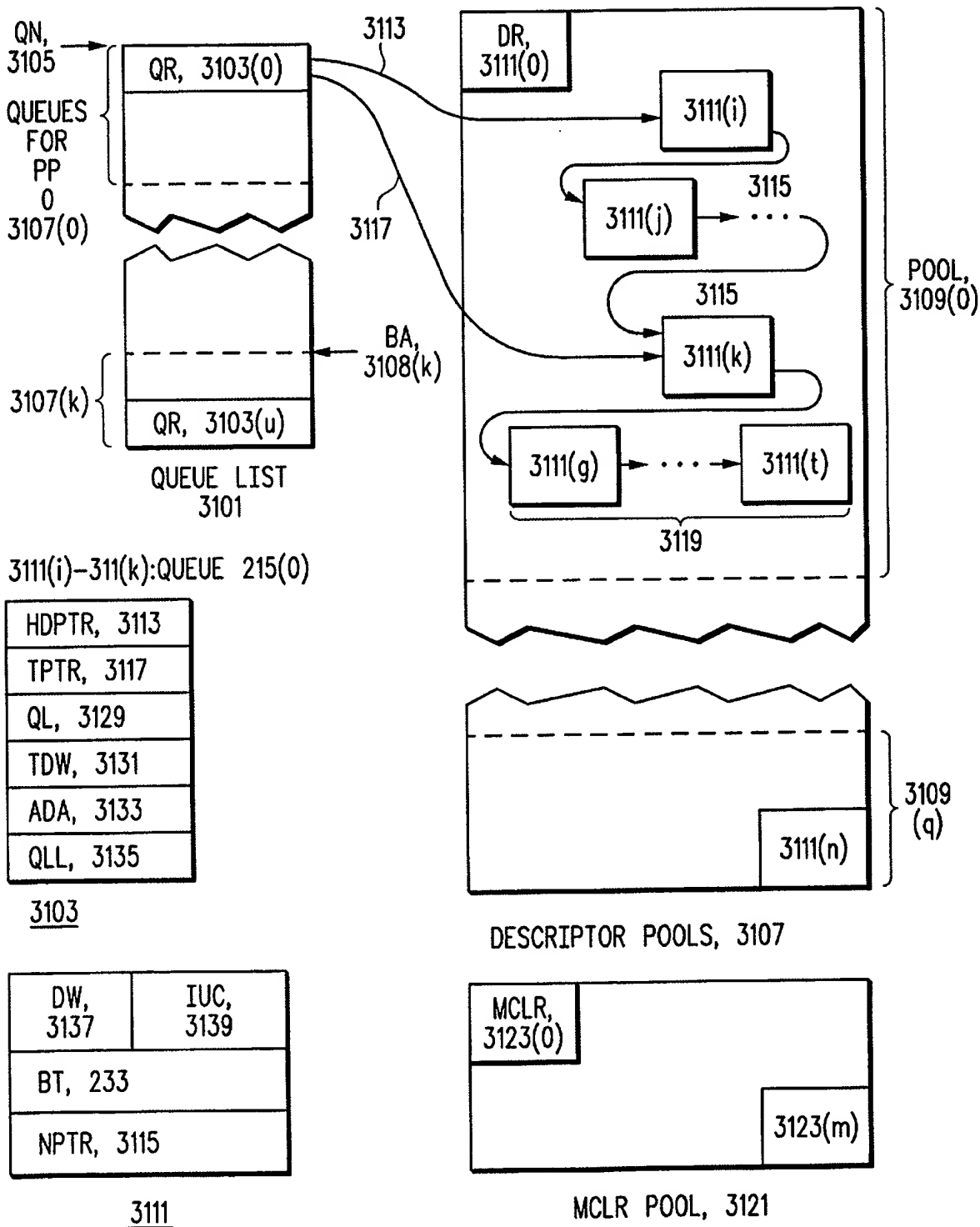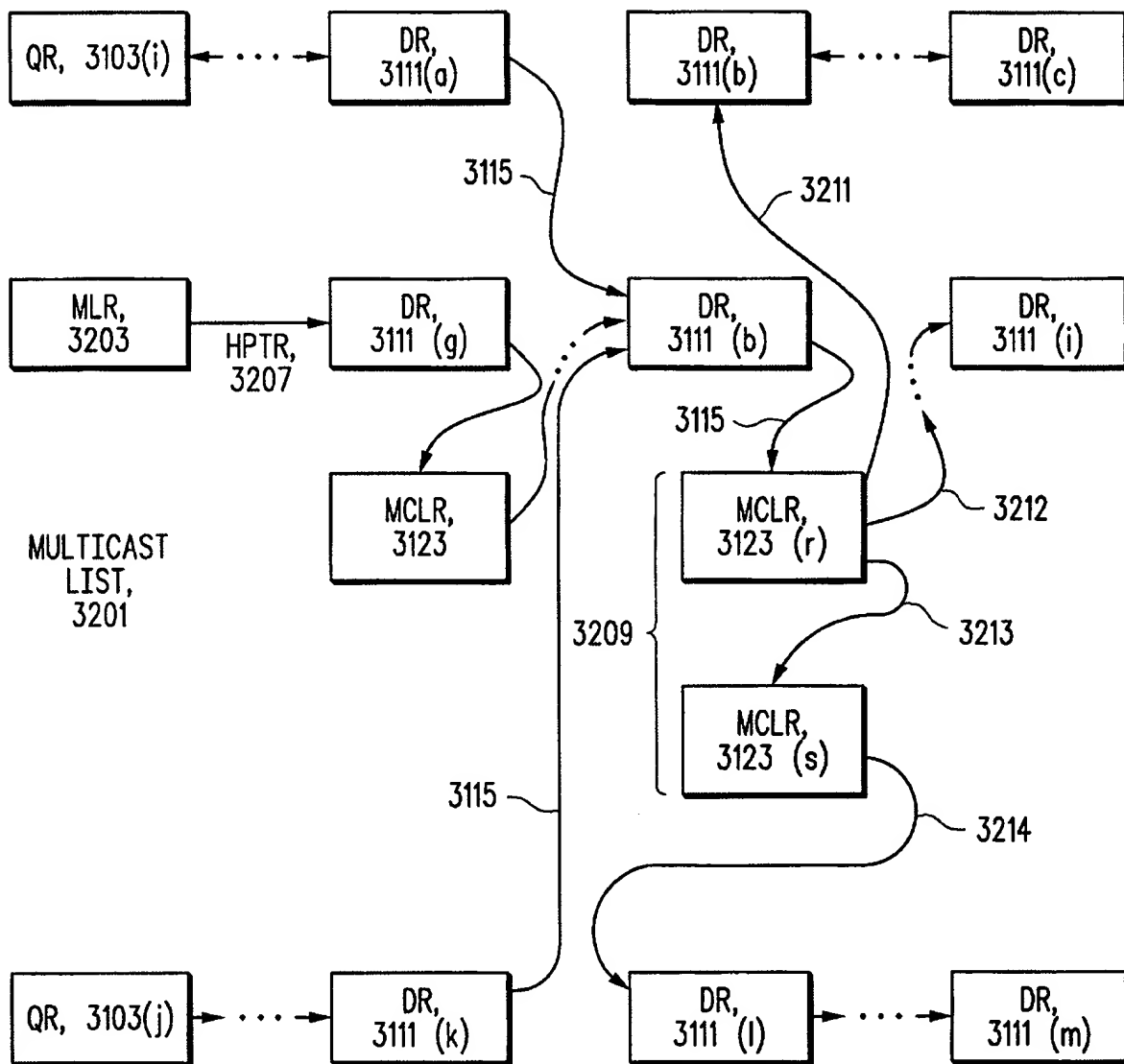
UNICAST QUEUE, 215(j)

Fig. 32

Replacement Sheet

QUEUE#, 3203	
NPTR, 3205	QPTR, 3301
3203	
3205	3301
NXTMCLR, 3213	

MCLR, 3123

USER #, QUEUING LEVEL#3309

3305

QNMTE, 3305		
USER (0)
PORTION, 3307(0)		

QROFF	NUM	(0,0)
NULL		(0,1)
NULL		(0,2)
QROFF	NUM	(0,3)
QROFF	NUM	(0,4)
NULL		(0,5)
NULL		(0,6)
NULL		(0,7)

3307(0)

3307(n)

QNMT, 3303

Fig. 33

Fig. 34

Replacement Sheet

QUEUE
MEMORY, 213

ADDR
0:19
3509

DATA
0:31
3503

CTL 0:4
3506

QUEUE, 305

MEMORY
EXTERNAL INTERFACE, 3501

D FLOW CTL
REG., 3515

EXTERNAL QUEUEING
SCHEDULE
UNIT, 3508

COMMAND
0:15, 3511

DATA
0:31
3503

CTL 0:7, 3513

RQ,
3519

S FLOW
CTL
REG., 3517

QME, 305

TQ,
3521
(0)

TQ,
3521
(24)

SCHEDULER
EXTERNAL INTERFACE, 3507

Fig. 35

Replacement Sheet

PINS, 3601   DIRECTION, 3603

3513 {
3605 → CLK   1 → DCP → SCHED

3607 → D_FLOW_CTRL   3 → DCP → SCHED
3609 → S_FLOW_CTRL   1 → DCP → SCHED

3611 → XFER_RQST   1 → DCP → SCHED

3613 → XFER_CTL   2 → DCP → SCHED

3511 {
COMMAND DATA   16 → DCP ↔ SCHED
CMD_PARITY   1 → DCP ↔ SCHED

3509 {
DESCRIPT DATA   32 → DCP ↔ SCHED
DATA_PARITY   1 → DCP ↔ SCHED

TOTAL   58

;IF = 0, THE SCHEDULER CAN ACCEPT AT LEAST ONE DESCRIPTOR.

;IF = 1, THE SCHEDULER HAS AT LEAST ONE DESCRIPTOR TO TRANSFER.

3507

FIRST COMMAND CODE, 3615
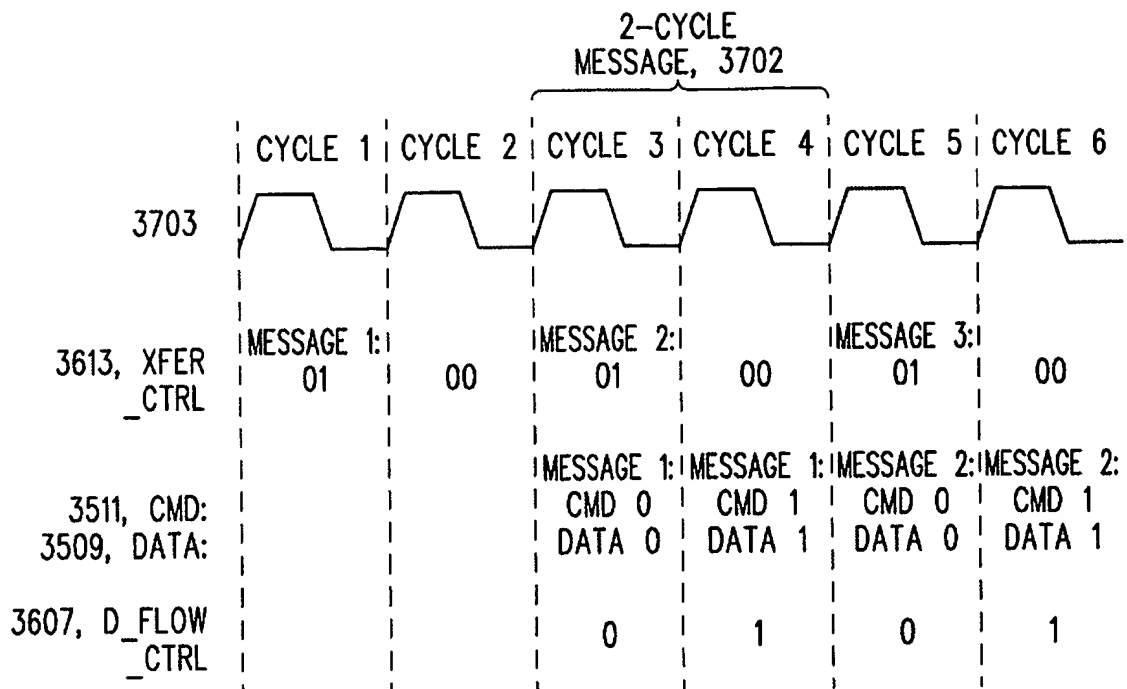
111111
5432109876543210
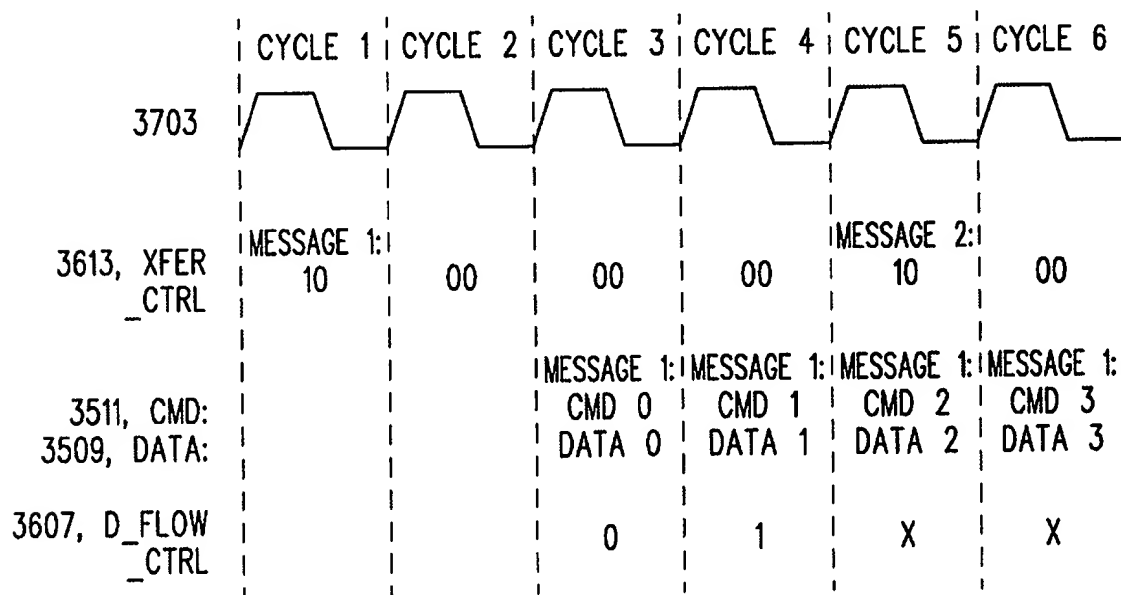
DESTINATION DCP PROCESSOR NUMBER, 3617

3514

Fig. 36

2-CYCLE
MESSAGE, 3702

	CYCLE 1	CYCLE 2	CYCLE 3	CYCLE 4	CYCLE 5	CYCLE 6

3703

3613, XFER_CTRL	MESSAGE 1: 01	00	MESSAGE 2: 01	00	MESSAGE 3: 01	00
3511, CMD: 3509, DATA:			MESSAGE 1: CMD 0 DATA 0	MESSAGE 1: CMD 1 DATA 1	MESSAGE 2: CMD 0 DATA 0	MESSAGE 2: CMD 1 DATA 1
3607, D_FLOW_CTRL				0	1	0	1

3701: 2 2-CYCLE MESSAGES

	CYCLE 1	CYCLE 2	CYCLE 3	CYCLE 4	CYCLE 5	CYCLE 6

3703

3613, XFER_CTRL	MESSAGE 1: 10	00	00	00	MESSAGE 2: 10	00
3511, CMD: 3509, DATA:			MESSAGE 1: CMD 0 DATA 0	MESSAGE 1: CMD 1 DATA 1	MESSAGE 1: CMD 2 DATA 2	MESSAGE 1: CMD 3 DATA 3
3607, D_FLOW_CTRL			0	1	X	X

3705: 2-4 CYCLE MESSAGES

4-CYCLE
MESSAGE, 3702

Fig. 37

Replacement Sheet

POOL
1D,
3819

BUFFER, 231(0, 0)

PDU, 2503

BT, 233

231(0, i)

OFFSET, 3802

231(0, m)

BUFFER
POOL,
3803(0)

BUFFER
POOL,
3803(n−1)

BTE, 3806

H,
3807

T,
3809

3806

BTQ
3805(0)

· · ·

BTQ
3805(n−1)

CNT,
3813

3819,
233

CNT

BT
COUNTERS,
3811

BT
POOL,
3803(n)

3801

ADDR, 3817   DATA, 3818

PB,
317

Fig. 38
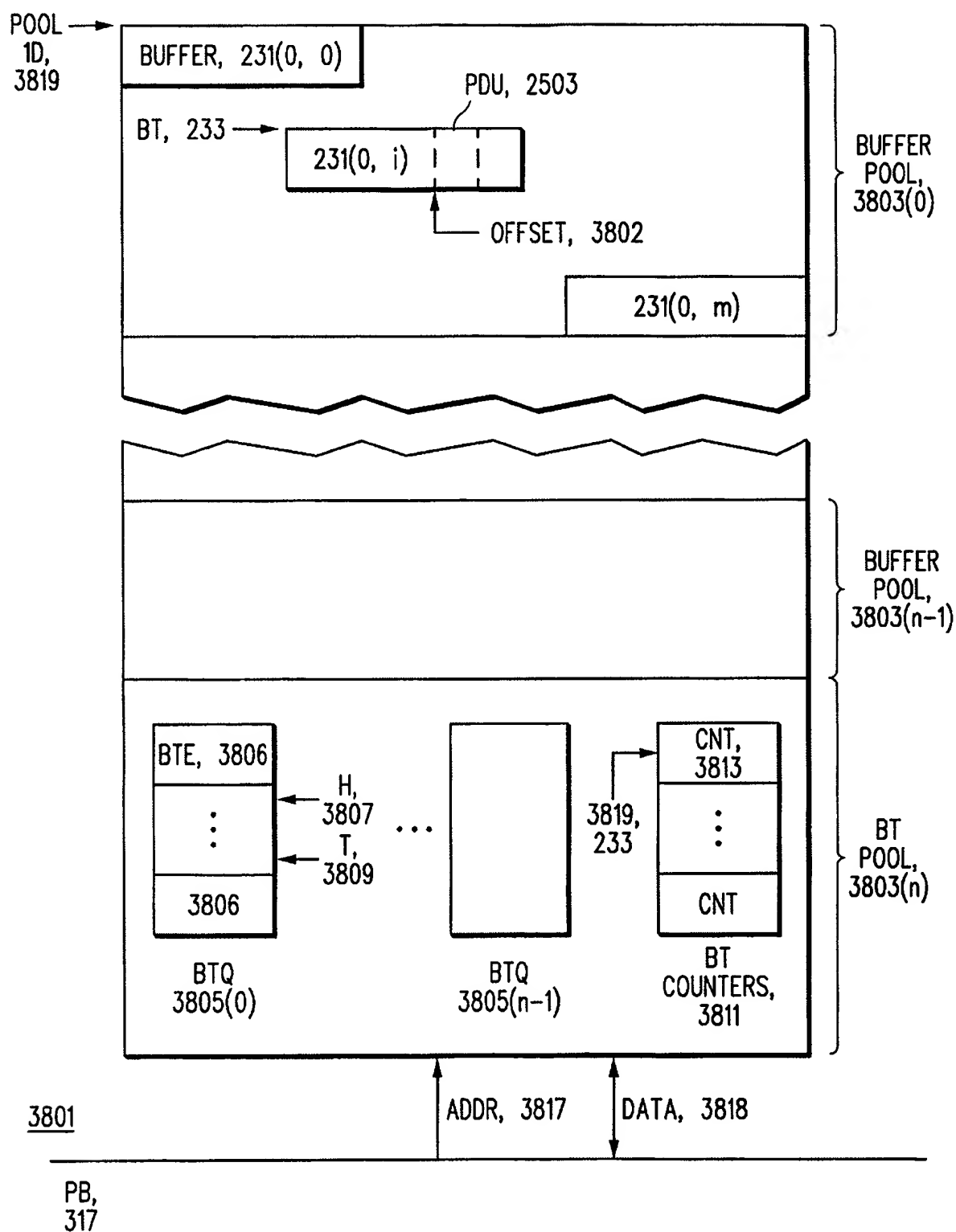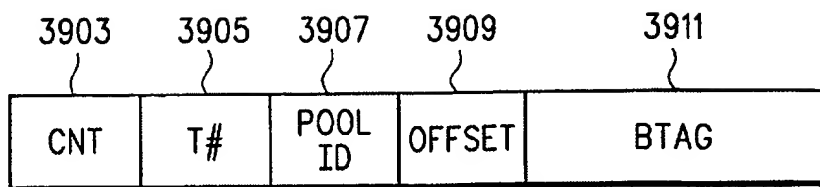
```
        3903      3905    3907    3909         3911
         ⌇         ⌇       ⌇       ⌇            ⌇
    ┌─────────┬────────┬───────┬────────┬──────────────┐
    │         │        │ POOL  │        │              │
    │   CNT   │   T#   │  ID   │ OFFSET │     BTAG     │
    │         │        │       │        │              │
    └─────────┴────────┴───────┴────────┴──────────────┘
                         └──────────────────────────┘
                                    BUFFER
                              ADDRESS, 3913
```

PAYLOAD BUS
BUFFER READ/WRITE COMMAND, 3901
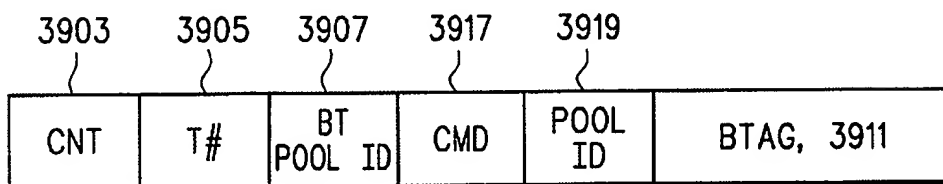
```
        3903      3905    3907    3917    3919
         ⌇         ⌇       ⌇       ⌇       ⌇
    ┌─────────┬────────┬───────┬───────┬───────┬──────────────┐
    │         │        │  BT   │       │ POOL  │              │
    │   CNT   │   T#   │POOL ID│  CMD  │  ID   │  BTAG, 3911  │
    │         │        │       │       │       │              │
    └─────────┴────────┴───────┴───────┴───────┴──────────────┘
                                 └──────────────┘
                                      3909
```

PAYLOAD BUS BTAG — COMMAND, 3915

Fig. 39

SC10981TS
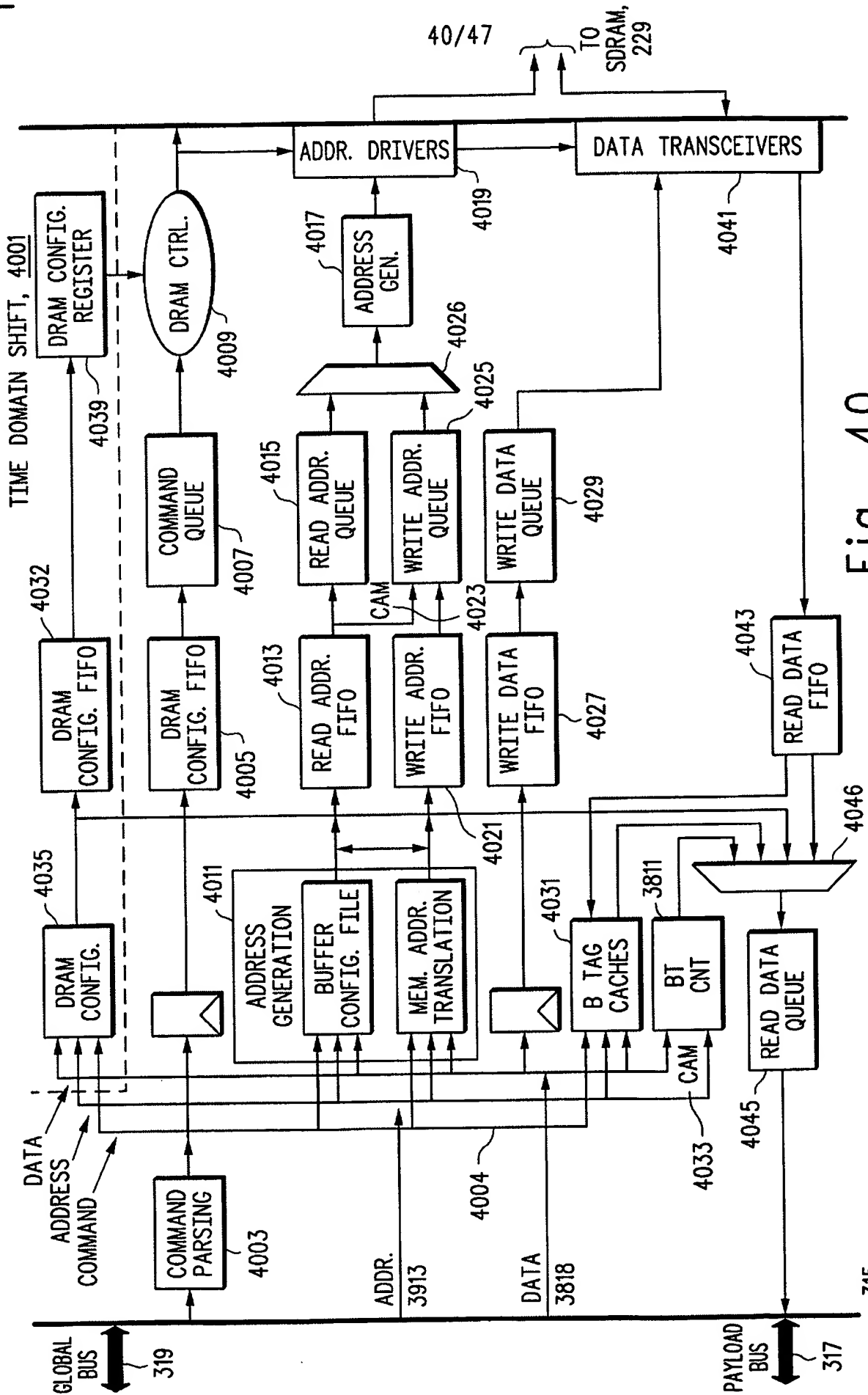
Fig. 40

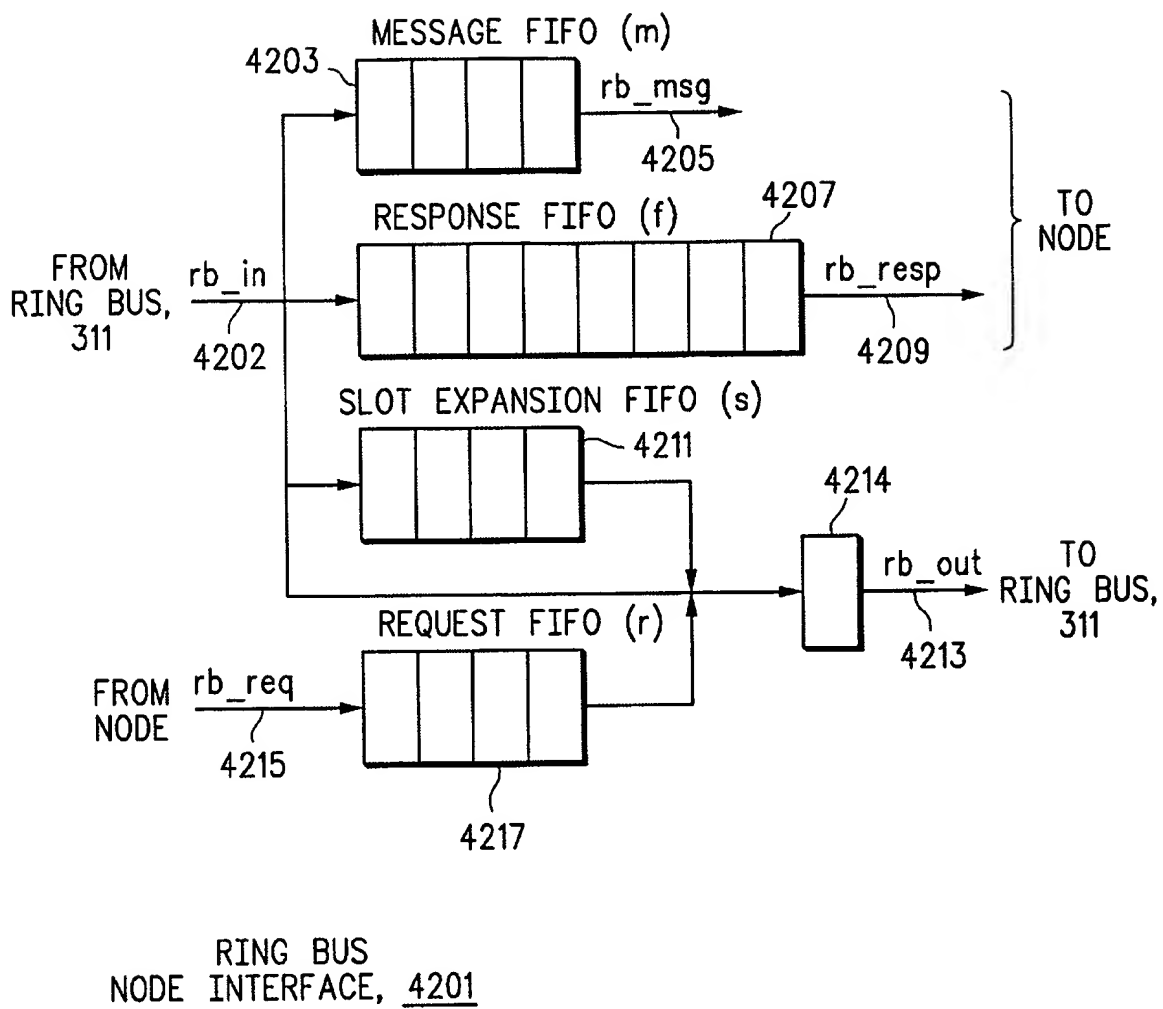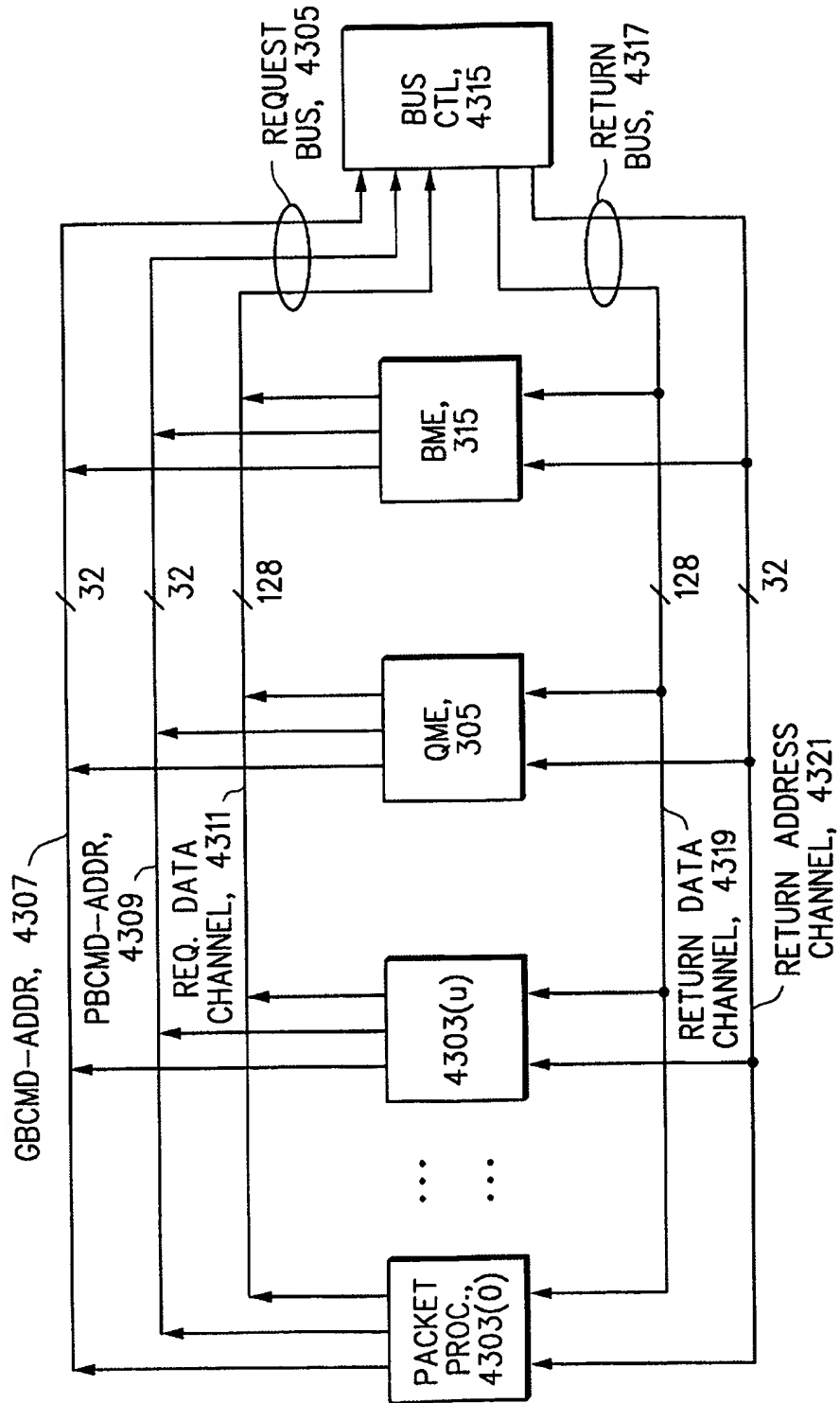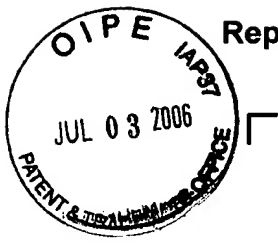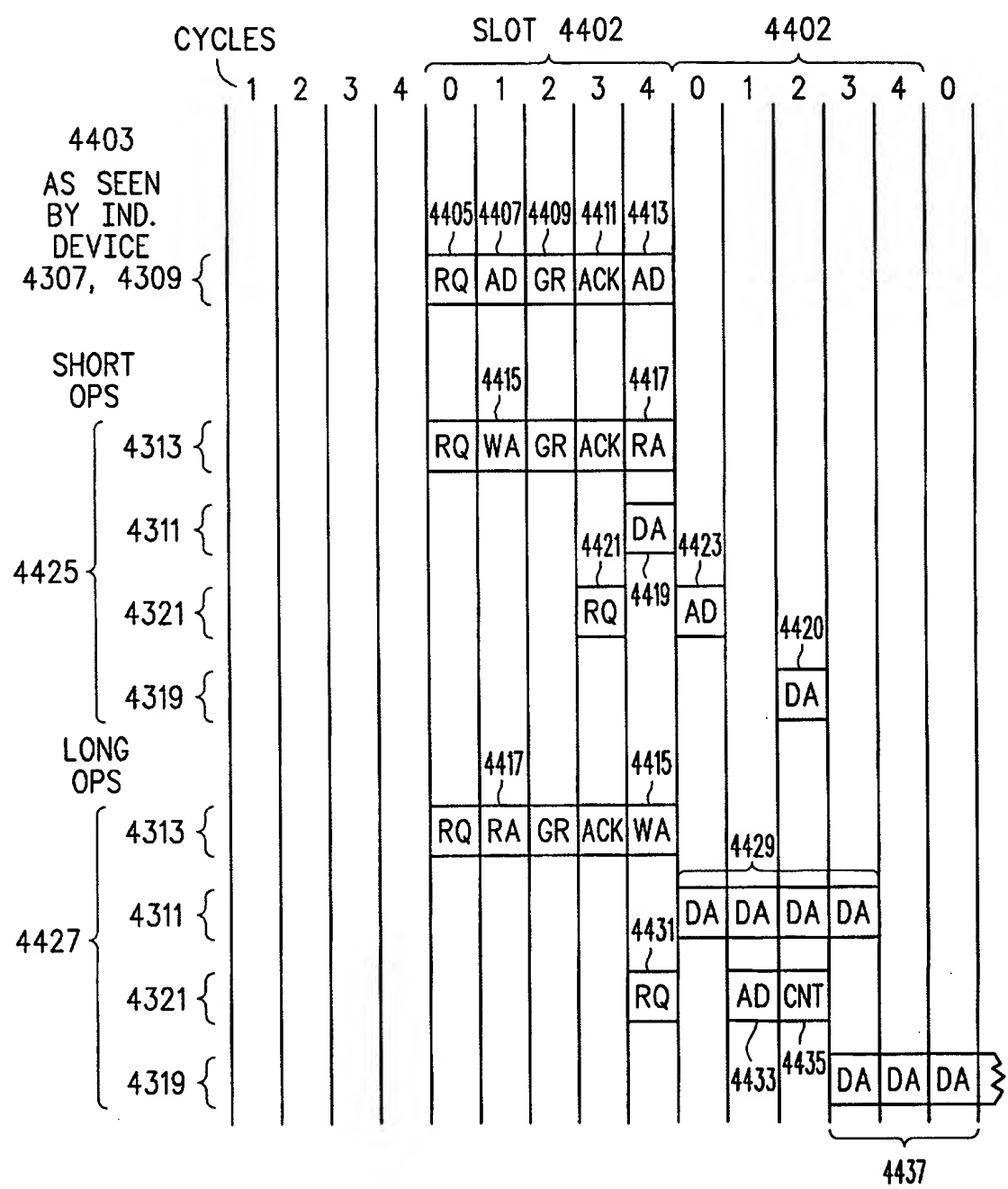| RTOS, 4101 |
| --- |
| BTAG AND BUFFER POOLS, 4103 |
| XP DATA MEMORY, 4105 |
| TRANSLATION TABLES, 4107 |
| PACKET PROCESSOR CODE AND DATA, 4109 |
| MEMORY CONFIG. INFO, 4111 |

229

Fig. 41

MESSAGE FIFO (m)

4203

rb_msg

4205

RESPONSE FIFO (f)

4207

TO
NODE

FROM
RING BUS,
311

rb_in

4202

rb_resp

4209

SLOT EXPANSION FIFO (s)

4211

4214

rb_out

TO
RING BUS,
311

4213

REQUEST FIFO (r)

FROM
NODE

rb_req

4215

4217

RING BUS
NODE INTERFACE, 4201

Fig. 42

Fig. 43

Fig. 44
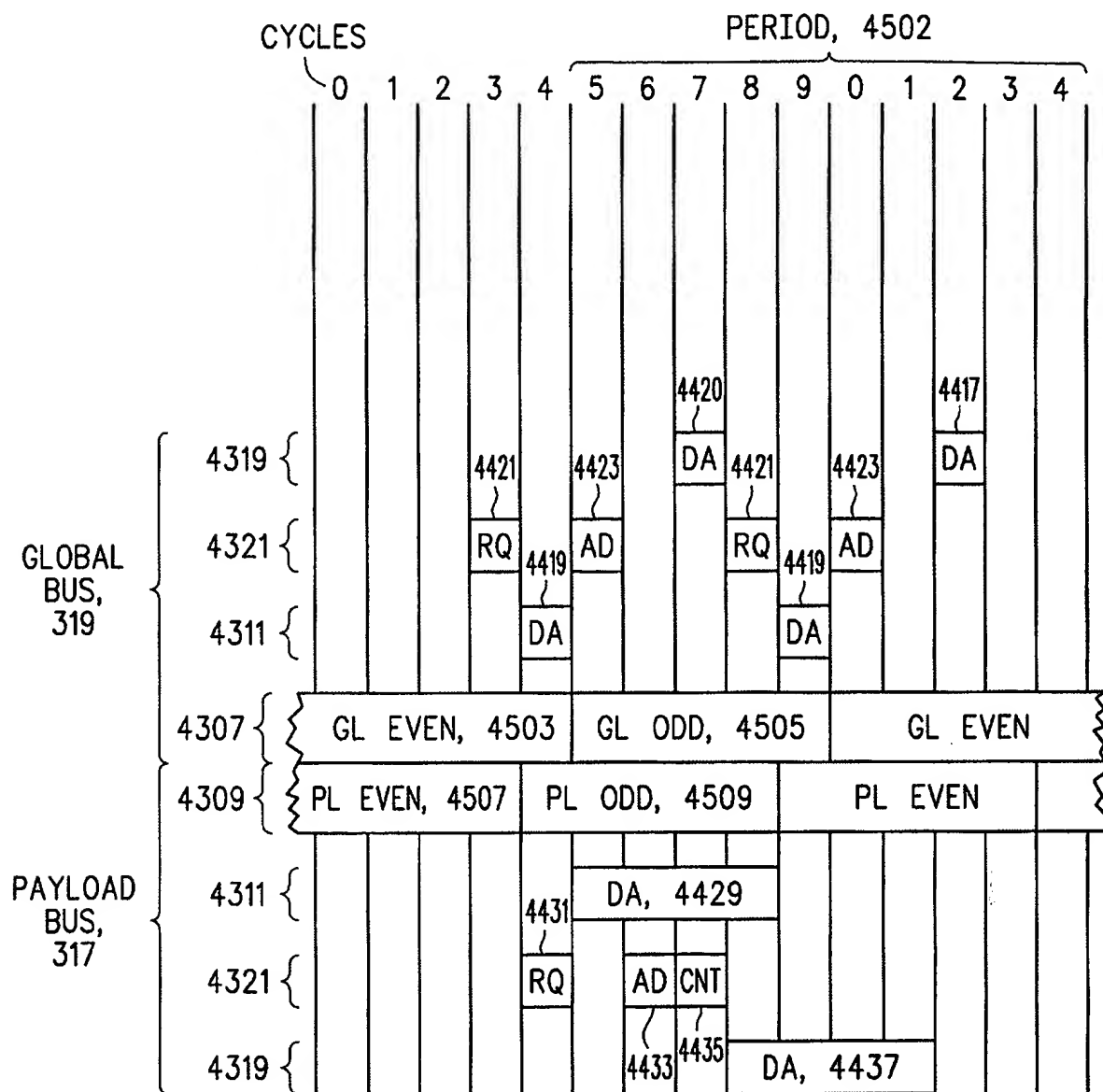
4501

Fig. 45

PIN	PURPOSE	RMII	OC-3	DS1	DS3	GMII (Tx)	GMII (Rx)	TBI (Tx)	TBI (Rx)	OC-12
CP0_0	OUTCLK	REF_CLK	RCLK_H	TCLK	TCLK	TCLK	nc	TCLK	nc	TCLK
_1	INCLK	CRS_DV	RCLK_L	RCLK	RCLK	CRS	nc		nc	TCLK1
_2	DATA	TXD(0)	TXD_H	TDATA	TDATA	TXD(0)	nc	TXD(0)	nc	TDX(0)
_3	DATA	TXD(1)	TXD_L	TxFRAME	TxFRAME	TXD(1)	nc	TXD(1)	nc	TDX(1)
_4	DATA	RXD(0)	RXD_H	RDATA	RDATA	TXD(2)	nc	TXD(2)	nc	TDX(2)
_5	DATA	RXD(1)	RXD_L	RxFRAME	RxFRAME	TXD(3)	nc	TXD(3)	nc	TDX(3)
_6	DATA	TX-EN	SIGNAL_DET			TX_EN	nc	TXD(4)	nc	
CP1_0	OUTCLK	REF_CLK	RCLK_H	TCLK	TCLK					
_1	INCLK	CRS_DV	RCLK_L	RCLK	RCLK	COL	nc			
_2	DATA	TXD(0)	TXD_H	TDATA	TDATA	TXD(4)	nc	TXD(5)	nc	TDX(4)
_3	DATA	TXD(1)	TXD_L	TxFRAME	TxFRAME	TXD(5)	nc	TXD(6)	nc	TDX(5)
_4	DATA	RXD(0)	RXD_H	RDATA	RDATA	TXD(6)	nc	TXD(7)	nc	TDX(6)
_5	DATA	RXD(1)	RXD_L	RxFRAME	RxFRAME	TXD(7)	nc	TXD(8)	nc	TDX(7)
_6	DATA	TX-EN	SIGNAL_DET			TX_ER	nc	TXD(9)	nc	
CP2_0	OUTCLK	REF_CLK	RCLK_H	TCLK	TCLK					
_1	INCLK	CRS_DV	RCLK_L	RCLK	RCLK	nc	RCLX	nc	RCLK	RCLK1
_2	DATA	TXD(0)	TXD_H	TDATA	TDATA	nc	RXD(0)	nc	RXD(1)	RDX(0)
_3	DATA	TXD(1)	TXD_L	TxFRAME	TxFRAME	nc	RXD(1)	nc	RXD(0)	RDX(1)
_4	DATA	RXD(0)	RXD_H	RDATA	RDATA	nc	RXD(2)	nc	RXD(2)	RDX(2)
_5	DATA	RXD(1)	RXD_L	RxFRAME	RxFRAME	nc	RXD(3)	nc	RXD(3)	RDX(3)
_6	DATA	TX-EN	SIGNAL_DET			nc	RX_DV		RXD(8)	FP
CP3_0	OUTCLK	REF_CLK	RCLK_H	TCLK	TCLK					
_1	INCLK	CRS_DV	RCLK_L	RCLK	RCLK			nc	RCLKN	
_2	DATA	TXD(0)	TXD_H	TDATA	TDATA	nc	RXD(4)	nc	RXD(4)	RDX(4)
_3	DATA	TXD(1)	TXD_L	TxFRAME	TxFRAME	nc	RXD(5)	nc	RXD(5)	RDX(5)
_4	DATA	RXD(0)	RXD_H	RDATA	RDATA	nc	RXD(6)	nc	RXD(6)	RDX(6)
_5	DATA	RXD(1)	RXD_L	RxFRAME	RxFRAME	nc	RXD(7)	nc	RXD(7)	RDX(7)
_6	DATA	TX-EN	SIGNAL_DET			nc	RX_ER	nc	RXD(9)	LOCKDET

4601

Fig. 46

	DATA CONFIG	RxCLK MUX	TxCLK MUX	RxDATA ENA	TxDATA ENA

4703   4705   4709

4701   4707   4711

	RX EN	RX BY EN	RX BI EN	RX SO EN	RX SY EN	RX BY LOOP	RX BI LOOP	AGG MODE	

4715   4721   4723   4725   4731

4717   4719   4729

4713

Fig. 47